RSTS-11 Sig DECUS

APRIL 1976 RSTS SPECIAL INTEREST GROUP NEWSLETTER VOL. 3 NO. 2.

FROM THE EDITOR

WE HAD QUITE A SIG MEETING IN DALLAS. I'M ONLY SORRY MORE OF YOU COULDN'T ATTEND. THE MINUTES OF THE MEETING ARE IN THIS NEWSLETTER. THE SUPPORT THAT ED CONVERSE, RSTS PRODUCT MANAGER, AND HIS PEOPLE GAVE US WAS JUST SUPER. THE 30 USERS WHO ATTENDED CAME AWAY WITH THE GOOD FEELING THAT THE RSTS MANAGEMENT WILL LISTEN TO THEIR USERS AND THAT THEY ARE TRYING VERY HARD TO PROVIDE US WITH A CONTINUALLY IMPROVING PRODUCT.

THE ONLY AREAS OF USER NEED THAT SEEM TO BE NEGLECTED ARE THOSE OF A GENERAL NATURE THAT HAVE TROUBLE FINDING FUNDING. AN EXAMPLE OF THIS IS THE LONG AWAITED RSTS INTERNALS MANUAL FOR USERS. IT IS SOMETHING THAT IS NEEDED, BUT CAN'T SEEM TO GET DONE BECAUSE NO AREA IN DEC WILL FUND IT.

THE SPRING DECUS SYMPOSIUM LOOKS LIKE THE BEST ONE EVER. A SCHEDULE OF THE RSTS SESSIONS IS IN THE NEWSLETTER. COME TO ALANTA!

BOB BRANTON
R LA BAR & ASSOCIATES
2750 NORTHAVEN RD., SUITE 102
DALLAS, TEXAS 75229
(214) 243-3451

RSTS V4

IT NOW APPEARS THAT A MEMBER OF THE RSTS V4B DEVELOPMENT GROUP WILL BE AT V4 SUBGROUP MEETING WEDNESDAY MAY 26 IN ALTANTA.

IN OTHER V4 DEVELOPMENTS, IT LOOKS LIKE THE VERSION 4 LISTING WILL BE OFFERED TO V4 USERS FOR A MEDIA PRICE. MORE ON THIS SHOULD BE KNOWN BY ALTANTA. ALSO THE RSTS SUPPORT GROUP AT DEC NOW HAS COPIES OF RALPH PRATT'S NOTES ON V4 AVAILABLE TO V4 USERS. WRITE THEM FOR COPIES.

ROY SWATZELL
V A HOSPITAL
ROOM 2 R12
BIRMINGHAM, AL 35233
(205) 933-8103 EXT 239

RSTS Sessions At Spring DECUS

The spring DECUS symposium will be at the OMNI International Hotel in Alanta from May 25 thru May 28. There will be more RSTS sessions than ever before. I have tried to list those sessions directly related to RSTS. In addition there will be many other sessions not listed here of general interest to you. For those users who are new to RSTS please note the new users session at 8:00 wednesday morning.

Tuesday - May 25th

1.	RSTS Product Panel (6B & cusps when, where, how)	Swanton Room 1:30 - 3:00
2.	RSTS Basic Coding Standards of users and inside DEC	Swanton Room 4:15 - 5:15
3.	PDP-11 File/Database Management Panel	Minosa Room 3:15 - 5:15

Wednesday - May 26th

4.	RSTS New Users Session	Swanton Room 8:00 - 8:30
5.	RSTS SIG General Session	Swanton Room 8:30 - 10:00
6.	RSTS SIG "WISH LIST"	Swanton Room 10:15 - 11:45
7.	RSTS Documentation Session	Swanton Room 1:30 - 2:30
8.	RSTS SIG Commercial Subgroup Session	Swanton Room 3:15 - 5:15
9.	RSTS SIG V4 Subgroup Session	Brampton Room 3:15 - 5:15

Thursday - May 27th

10.	DMS-500 Database Management	Swanton Room 8:30 - 10:00
11.	RSTS DBMS Paper a. Detached ISAM under RSTS b. General Application Monitor under RSTS c. Large Database Management under RSTS	Swanton Room 10:15 - 11:45
12.	RSTS Informal Discussion (DEC)	Swanton Room 3:15 - 5:15

Friday - May 28th

13.	RSTS Application Papers a. Structured Programming under RSTS b. Metarelational Control of Relational Databases c. Improvement of RSTS/E Performance	Swanton Room 8:30 - 10:00
14.	DECAL and CAI on RSTS/E	Westover Room 10:15 - 11:45

BASIC-PLUS and RSTS Programming Standards Session at Atlanta

A special user session has been set aside at the Spring DECUS meeting in Atlanta, to discuss the subject of RSTS-11 and BASIC-PLUS programming standards.

We are going to try to have speakers participating in the session to cover the following topic areas:

New RSTS/E CUSP Programming Standards (DEC personnel)

Some thoughts on areas of concern in programming Standards

One or more user experience presentations

As I am the current area chairman for programming standards within the RSTS-11 SIG, I want to use the session at Atlanta to get a dialogue going between members of the SIG and DEC as well. To that end, I will formulate a questionnaire, for user participation, concerning discussion areas you might be interested in. We are also going to attempt to establish a column in the SIG newsletter, appearing on a regular basis, where we can present various viewpoints on the issues, and give tutorials to new users who desire help in getting started.

I am currently of the opinion that programming standards should be formulated in such a way as to serve as guidelines to people using RSTS and programming in BASIC-PLUS. As a seller of package software, I know the benefits to be gained for standards in programming within an organization. Communication between users, such as through the DECUS library, would also benefit from an accepted set of programming conventions and standards.

I also believe that RSTS and BASIC-PLUS have matured to the point that they are generally accepted as viable computer technology components and no longer just "toys". Therefore, I know that you out there have much to contribute in a dialogue on programming standards and conventions.

Lets get together at Atlanta then, and get a dialogue going, one which I am sure will benefit all of us both old and new users of RSTS and BASIC-PLUS. Both new BASIC-PLUS II language and the Version 6B release of BASIC-PLUS have new features enabling us to write more readible code and more modular programs. Now is the time to start preparing ourselves for the next generation.

MINUTES FOR THE MARCH 13 AND 14 - DALLAS RSTS-SIG

This meeting was co-chaired by Mr. Bob Branton, chairman of the RSTS-SIG and Stuart Katz, chairman of the Commercial RSTS SUB SIG. There were approximately 35 people in attendance and DEC was well represented from Maynard. People attending were:

Jim Bivins
Deena Boone
Jack Bouknight
Bob Branton
Elinor Burns
Mike Carr
Ed Converse
Al Cordes
Gary Cranson
Gene Dugger
Ray Frank
Tom A. Hamilton
John Hammond
Mark Hendricks
Mary Ann Hodges
Ken Hogan
Ed Hopey
Gerry Hornik
Phil Houghton
Dwaine Jones
Bob Jones
Burton Katz
Stuart Katz
Mike Moore
Dian Pekin
Bill Picott
John Ritzenthaler
Joan Salkin
Bill Sconce
Murray Stinson
Jeff Wells
Bryan Weyrick
Harvey Wiggins

The meeting began with a review by Mr. Bob Branton regarding the history of this meeting and the primary reasons for developing a "wish-list" for future developments by DEC as discussed at the L.A. meeting.

Mr. Ed Converse from DEC then proceeded to explain DEC's goals and some of DEC's philosophies and then introduced Mr. Ed Hopey, who is responsible for the development of RSTS, who in turn gave us a review of RSTS Version 6B. Particular items that were of interest were as follows:

1. RSTS Version 6B will be self-generated. No DOS will be required.
2. Error logging will be enhanced.
3. Bad block handling will be enhanced.
4. Backup will be enhanced per the discussions of L.A. (This topic was tabled until later in the day for greater detail.)
5. User environment test package to be developed to test both hardware and software problems.
6. Auto-patching will be available. All patches will be made directly through command files as released by DEC rather than user input.
7. A new spooler operator control program. This program will communicate with all spoolers, batch, etc., to give the operator better control of these detached jobs.
8. The CUSPS are being reviewed and enhanced.
9. Language enhancements being made include - 32 character variables, unlimited lines and the "&" will be used to extend the line, therefore "&" can no longer be a substitute for the word "print" and imbeded comments.

NOTE →

Mr. William Picott, product manager for Central Engineering Languages, then discussed Basic Plus II. It was indicated that Basic Plus II was not a replacement for Basic Plus but will coincide with Basic Plus. The goals of Basic Plus II are:

1. Execution speed. Five to ten times faster.
2. Language enhancement to include ISAM, record management, all extensions from Basic Plus, i.e., print using, matrixes etc.
3. Provide additional space for programs.
4. In core compiler.
5. Basic Plus compatibility. The real goal here is to make Basic Plus II compatible with the following - Basic Plus, Basic-11 and Basic-20. The major difference in languages will revolve around record management systems heretofore known as RMS. Record I/O will be available under Basic Plus II but RMS will not be available under Basic Plus.

A translator will be provided to convert Basic Plus programs to Basic Plus II.

Most of us are concerned about the immediate mode not being available within the language in Basic Plus II. We will not be able to provide expressions during execution for de-bugging. We will be able to print variables. We will also be able to assign variable values and we will be able to stop programs as necessary. We will not be able to insert new lines or print expressions.

Mr. Picott then questioned why Cobol had not caught on better with RSTS users. The two major reasons suggested were that ISAM and external CALLS are not available. Without these two items, the users do not feel that Cobol is a viable product. The second release of Cobol will not provide either ISAM or external CALLS, but the third release will. A compiled Cobol will be available with the third release. The topic of Dibol under RSTS was approached. The tally was taken that a half dozen of the 25 users present would like to use Dibol but most of the others were not familiar with its capabilities. It is suggested that Dibol will be available around the time of 6B.

Mr. Gerry Hornik, the Development Manager of Large Business Systems, with concentration on RSTS, was then introduced and discussed in further detail the release of Dibol and RPG. Dibol should be released around 6B and RPG, within the next four months. He explained some of the reasons for going to an RPG, pointing out that it would be IBM System Three, RPG II compatible as much as possible. One of the major problems that DEC is attempting to overcome with the RPG is providing a good text editor, since RPG is a predetermined format, fill in the blank programming language. Mr. Hornik went into detail explaining RMS. Most of those users present were most interested in this product and it appears that future releases would utilize RMS-11 as an option. We hope to hear more of RMS at the next DECUS convention in Atlanta.

After lunch, Mr. Ed Hopey was called upon to give us an extended discussion on the new Backup since this was of great concern to most users at the last DECUS convention (L.A.). The following is his review of the basic points that the new backup covers:

1. Backup uses non-file structured output. It is not RSTS format and can only be utilized through backup.
2. User has Wild Card capabilities within the file names, extensions, and account.
3. User can call for backup based upon dates.
4. Users can use an "except" clause i.e., except *TMP files would print all files except *TMP files.
5. A command file can be used instead of input from the user.
6. There is a begin-at clause so that the user does not have to start at the beginning of each account or file, etc.
7. Users can delete files upon completion of the backup.
8. An orderly abort command is available.
9. A pause command is available.
10. A continue commandis available.
11. The capability of determining the current status is available.
12. Backup will run detached.
13. It will run faster.
14. It will utilize index files, therefore the sequence for backup is:
    a) Dialogue phase.
    b) Selection and directory index creation.
    c) Transfer phase.
15. Backup from account to account is allowed.
16. Superceded switch still exists.
17. There is no verification with this backup.

This new version of backup will be released with Version 6B. At this point, the question existed as to whether backup would be able to read files that had previously been locked or in the update mode. There was no agreed concensus of the group but it was the feeling that the least dangerous would be to allow backup to read everything. It appears as if it is necessary to find a third mode to open a file as "read only" and read everything, whether it is opened for input or opened for mode one update.

Further discussion of Basic Plus II was held with the group addressing most of the questions to Gerry Hornik. It was noted that the Basic Plus II would have different print using formats more compatible with the Dartmoth Basic and this might create the only problem in the translation from Basic Plus to Basic Plus II. It was also stated that a trace routine would be available for de-bugging to trace the line numbers being executed. It does not give values of the variables maintained. It is anticipated that version 1 of Basic Plus II will be available November 15th.

The discussion then led in the direction of the current CUSPS. The first area covered was that of editors, including of TECO and other CRT editors. To date, TECO will only be released through the DECUS library and RSTS users should go there if they are interested in using TECO. We did suggest to DEC that the most important factor of any editor would be that it should be a CRT rather than a line editor. We then opened up discussion on the batch, spool, Que and Que-manager programs. We found that the major question regarding these programs had been answered by the new operator control program to be released as part of 6B. The discussion did go into how the batch system works and it was felt that further information should be sent to DEC in order to assist them in plans for any future growth of batch

Mr. Hornik gave a short report on the status of his ISAM which will be available approximately May 15, 1976. It should be noted that he gave a paper in L.A., for this reason he did not go into detail regarding his current project.

Questions regarding error logging, error copy and error display were raised and it was found that certain users were having problems in distinguishing which errors were or were not being caught. It was also suggested that a program, possibly called CRUNCH, should be developed in order to analyze which user is putting the heaviest load on a system at any one time.

Late afternoon Saturday, we were also given a review of the up-and-coming schedule for the Atlanta conference with specific times and subjects given for the RSTS sessions. We will have 3½ days of solid RSTS sessions scheduled so the meeting should be as beneficial as possible. We also discussed a variety of hardware requirements such as remote warning printers, remote line printer drivers and various modems needed by users. We also spent some time reviewing the "wish-list" suggestions from the day and then adjourned the meeting until Sunday morning.

Sunday morning we met and reviewed the "wish-list" as we felt it should be prepared and directed Mr. Bill Sconce to prepare another questionaire for the next RSTS newsletter in an attempt to get a survey completed prior to the RSTS meeting in Atlanta. We also opened the subject of documentation and software support to the floor. Elinor Burns from DEC Software Maintenance reviewed the software maintenance agreement and procedures with us and discussed the variety of problems which we have all witnessed. The meeting was wrapped up with thanks extended to the DEC personnel who took the time to join us, and members of SIG who had the opportunity to come down to Dallas for making the meeting as successful as it was. The meeting provided a

solid basis from which the "wish-list" session at the Atlanta DECUS meeting can work, therefore helping to make the session as profitable and advantageous to all users as possible. We look forward to seeing everyone at the DECUS meeting in Atlanta.

For further information regarding the Dallas meeting, contact any one of the members who attended.

NAME	ADDRESS
Ed Hopey	70 Hunt Rd., Chelmsford, Ma.
Joan Salkin	806 Taurus Drive, Foster City, CA.
Mike Carr	3058 Yale Blvd., Dallas, Tex.
Phil Houghton	Dr. Pepper, Dallas, Box 5086
John Hammond	" " "
Ed Converse	DEC
John Ritzenthaler	3830 Lakebriar, Boulder. Col.
Gene Dugger	102 Chrisp, Searcy, Ark. 72143
John Nunnally	Box 744, Searcy, Ark. 42143
Mark Hendricks	1966 Inwood Rd., Dallas 75235
Harvey Wiggins	" " "
Mary Ann Hodges	DEC-Dallas
Tom A. Hamilton	Louisiana-soon to be Okla City
Burton Katz	401 W. Fullerton, Chicago
Stuart Katz	401 W. Fullerton, Chicago
Deena Boone	401 W. Fullerton, Chicago
Bryan Weyrick	1200 E. Main, #42, Edmond, Okla.
Dwaine Jones	10511 Evesborough, Houston, Tx
Al Cordes	9039 Katy Frwy. #404, Houston
Ken Hogan	DEC-Dallas sales
Jeff Wells	4425 Gilbert Ave., Dallas, Tx.
Bill Sconce	U of Louisville HSCC; Box 1055
Jack Bouknight	PLYCOM Services, Box 160, Plymouth
Jim Bivins	R. LaBar & Assoc., 2750 Northaven
Bob Jones	Interactive Info., Box 37408, Cincinnati, Ohio
Murray Stinson	Scientific Placement, Inc., 2700 S. Post Oak, Houston 77027
Ray Frank	Environmental Developers, Inc., 13693 E. Iliff Ave., Denver, Col.
Mike Moore	Compucon, Inc., 13749 Neutron Rd., Dallas 75240
Gary Cranson	Southeast Colo. Power Assn, Box 521, La Junta, Col.
Dian Pekin	Digital Equip. Corp., Main St., Maynard, Ma.
Gerry Hornik	Digital PK3/1/M33, Maynard, Ma.
Bill Picott	Digital ML12-3/E13, Maynard, Ma.
Bob Branton	L. Labar & Assoc., Dallas
Elinor Burns	DEC, 146 Main St., Maynard, Ma.

RSTS/E BATCH PROCESSING

As a large commercial user of RSTS/E and after having worked with several commercial languages under RSTS/E, we are very irritated with DEC's implementation of a Batch processing facility. We think the reason for this implementation is because DEC has traditionally been a terminal oriented company and hasn't looked at Batch from a realistic working environment, therefore, didn't know how to implement batch. The introduction of batch languages under RSTS/E implies that DEC is serious about the commercial market. This article is intended as a caution to older DEC users and to potential new customers of batch languages about pitfalls in the batch processing area of RSTS/E, and to suggest some potential solutions to these problems that must come from DEC. Traditionally, reversing a decision of the vendor on a product requires a great deal of time because most of the time the minds that must be convinced of the new criteria are the same people responsible for the first product. Expecting this to be the norm, a strong reaction from the user community will be necessary before any change will be made in the product.

Our installation is a typical shop with outside terminals doing on-line data collection, verification and information retrieval. The following guidelines apply to the batch environment:

(a) There must be multiple batch streams executing simultaneously to accomplish the work load. This implies the following:

(1) Full-time operator loading, monitoring and executing jobs.

(2) A quality control area organizing input job streams and distributing outputs. (We feel that certain information should be restricted from user areas and kept in the confines of a highly secure area. Because of this opinion our batch area has a definite and distinct function which will not disappear with time.)

(b) All batch jobs to be executed must have proper JCL (including programmers' test runs). The batch operator must have control of all peripherals relating to batch production. However, this doesn't preclude programmers from queing to specific batch devices, but they cannot interfere with normal batch production, and must be controlled by the batch operator.

The following are problems that we have encountered during system development:

(a) Unnecessary operator intervention is required because of inter-action between 'Que' and 'Queman'. An operator must run 'Que' to execute a batch stream. He must then re-log into the account and attach to 'Queman' in order to respond to messages. After which he must detach and rerun 'Que' to monitor the batch ques.

(b) 'Batch' automatically ques JCL listings to the spooler. This is not initiated by the operator and results in uncontrolled interspersed printing.

(c) When attached to 'Queman' it is necessary to type the '/ME' switch to receive additional messages.

(d) The following are problems encountered with the line printer spooler:

(1) Automatic top-of-forms when job begins causes mis-alignment of pre-printed forms.

(2) The 'no-header' switch gives one page of undesirable header output and may cause mis-alignment of forms. (Pun - we wonder if DEC prints their paychecks under RSTS/E Batch.)

(3) When attached to 'Queman' and printer is off-line, and you respond to 'off-line' message with 'CO', a large 'OPERATOR CONTINUED JOB' message prints in the middle of spooled output.

(e) There doesn't seem to be a way to run multiple Batch streams.

(f) The JCL set does not include logical to physical device assignments. The print control statement is incomplete in that there is no forms control nor mode selection.

We feel the following things are necessary for batch operations:

(a) The batch operator should be able to enable que requests, display ques, receive and respond to messages, hold and release any/all qued batch and spooling jobs, alter priorities, rearrange jobs

in ques, issue restarts and multi-copies, assign and de-assign devices from one dedicated job. This job could be running on the system console terminal.

(b) The JCL should be logged on a dedicated hard copy device.

(c) The ability of other users to que jobs to input and/or output ques without interruption of batch processing and under batch operations control.

(d) JCL should include the following features which are not presently in batch:

(1) Forms document number and length

(2) Que priority control

(3) Logical to physical device assignments between all languages

(4) Multi-copies without multi-que request

(5) Imbed data within JCL

(e) The print spooler should recognize different form control numbers, issue the appropriate message to the batch operator, halt the line printer, and wait for operator response. Also needed is page restart.

(f) The ability to run multiple batch streams.

We presently have a similar system running in RSTS/E Basic Plus. Ideally it should be at monitor or run-time system level and supported by DEC.

Bryan Weyrick
Computer Utilities Corp.
4900 N. Portland
Oklahoma City, Okla.

Chuck Metcalfe
Houston Belt & Terminal Ry. Co.
Texas & Crawford
Houston, Texas 77002

Dwaine Jones
Computer Utilities Corp.

At the RSTS/E SIG meeting in Dallas, we determined that not very many users are making much use of the inter-task send/receive facilities of RSTS/E. We did not determine the reason for this, but this article is for those of you who are holding back because you are not sure just how to make two programs talk to each other.

First, inter-task send/receive does work. I have been using it extensively for over two and one-half years now, and I think I have a reasonably good feel for how to use it. I have based an entire data base manager on send/receive and have achieved quite acceptable response time.

Second, send/receive is not hard. The function I have included with this article is the result of a good deal of experimentation and is designed to enable easy and accurate communication between two or more RSTS/E jobs.

Third, the function uses a protocal which is compatible with the RSTS/E "QUEMAN" program's message protocal. It proves to be very important that a standard protocal be established to avoid problems when the receiving task tries to reconstruct the message. The protocal used in the "FNM" function is simple, accurate, and extremely flexible.

In order to send a message to another program, a program must have declared itself to be a receiver. It must declare itself as a unique receiver name. This name can be any valid file name up to six characters long, and is stored in rad-50 form. If the name is already in use by another task, an error 18 will be returned. if there are no more receiver slots available, an error 4 will be returned.

So step one, is to set up the receiving program:

```
100 INPUT "RECEIVER NAME";R$                              ! GET RECEIVER NAME
110 R$=MID(SYS(CHR$(6%)+CHR$(-10%)+R$),7%,4%)             ! CONVERT TO RAD-50
120 INPUT "HOW MANY BUFFERS";B%                           ! NUMB OF SMALL BUFFERS
130 INPUT "PRIVELEGED (1) OR NOT (0)";P%                  ! PRIVELEGED ONLY??
140 P%=128% IF P%<>0%                                     ! SET PRIVELEVED BIT
150 Z$=FNM$(0%,0%,"","")                                  ! TURN OFF RECEIVER
160 Z$=FNM$(1%,P%+B%,R$,"")                               ! TURN ON RECEIVER
170 GOTO 200 IF Z$=""                                     ! CHECK FOR ERROR
180 PRINT "ERROR";ASCII(Z$);"IN DECLARE"                  ! PRINT ERROR CODE
190 GOTO 100                                              ! TRY AGAIN
200 Z$=FNM$(2%,P%+B%,R%,"")                               ! NOW STALL FOR MESSAGE
210 PRINT "MESSAGE OF LENGTH";LEN(Z$)-1%;"FROM JOB";ASCII(Z$)/2%
220 PRINT "'";RIGHT(Z$,2%);"'"                            ! PRINT THE MESSAGE
230 GOTO 200                                              ! GET NEXT MESSAGE
```

Once the receiving program has been set up, the sending program can try to send a message to it. In order to send the message, you must know either the job number of the receiver program, or the name of the receiver declared by that program. The message can be sent to either the receiver name or the job number.

The receiving program does not need to be waiting for a message at the time one is sent, but it should be prepared to check the receiver queue

pretty quickly.

Here is a simple program to send a message:

```
100 INPUT "NAME OF RECEIVER TO SEND TO";R$          ! GET RECEIVER NAME
110 R$=MID(SYS(CHR$(6%)+CHR$(-10%)+R$),7%,4%)      ! CONVERT TO RAD-50
120 INPUT "MESSAGE";M$                             ! GET MESSAGE TEXT
130 Z$=FNM$(-1%,0%,R$,M$)                          ! SEND MESSAGE
140 GOTO 100 IF Z$=""                              ! CHECK FOR ERROR
150 PRINT "ERROR";ASCII(Z$);"IN SEND"              ! PRINT ERROR CODE
160 GOTO 100                                       ! DO IT AGAIN, SAM
```

Of course, if the sending program wants to receive a reply to its message, it must declare itself as a receiver, and so through the receive logic shown in the first program.

Here is the text of the FNM function. the function can be easily modified into several separate subroutines for high volume applications.

```
20300!
!
!       FUNCTION TO PERFORM INTER-TASK COMMUNICATION
!
!       P% = -1 FOR SEND, 0 FOR CANCEL, 1 OR 2 FOR RECEIVE
!       J% = JOB   FOR SEND & CANCEL, MSG COUNT FOR RECEIVE
!       Q$ = QUEUE NAME
!       T$ = TEXT FOR SEND, ELSE NULL
!
20301!
!
!       THIS FUNCTION WILL PERFORM ALL LOW VOLUME INTER-TASK
!       COMMUNICATION  FUNCTIONS.  IT SHOULD BE REWRITTEN AS
!       SEPARATE SUBROUTINES FOR HIGH VOLUME APPLICATIONS.
!
20310   DEF FNM$(P%,J%,Q$,T$)
        : DIM Z1$(63%) ! ARRAY USED TO STORE MESSAGE PACKETS

20320   Z1$=T$+""
        : Z2$,Z3$=""
        : Z4$=Q$+SPACE$(4%-LEN(Q$))
        : Z8%,Z9%=0%
        : ON ERROR GOTO 20360

20330   IF P%<>-1%
        THEN Z1$=""
        ELSE Z%=LEN(Z1$)
        : Z%=255% IF Z%>255%
        : Z3$=CHR$(Z%)
        : IF Z%>19%
        THEN Z2$=RIGHT(Z1$,20%)
          : Z1$=LEFT(Z1$,19%)

20340   Z$=SYS(CHR$(6%)+CHR$(18%)+CHR$(P%)+CHR$(J%)+Z4$+Z3$+Z1$)
        : IF P%=-1%
        THEN IF Z%<20%
```

```
            THEN 20380
            ELSE Z1$=Z2$+""
            : Z8%=0%
            : GOTO 20330

20350    GOTO 20380 IF P%<1%
         : Z%=ASCII(MID(Z$,5%,1%))/2%
         : Z0%,Z1%=ASCII(MID(Z$,9%,1%))
         : Z1%=19% IF Z1%>19%
         : Z1$(Z%)=Z1$(Z%)+MID(Z$,10%,Z1%)
         : GOTO 20340 IF Z0%>19%
         : FNM$=CHR$(Z%*2%)+Z1$(Z%)
         : Z1$(Z%)=""
         : GOTO 20390

20360    RESUME 20370 IF P%=1% AND ERR=5%
         : RESUME 20340 IF P%=2% AND ERR=5%
         : IF P%=-1% AND ERR=32%
         THEN SLEEP Z8%
           : Z8%=Z8%+1%
           : RESUME 20340

20361! SLEEP FOR SUCCESSIVELY GREATER INTERVALS UNTIL BUFFERS BECOME
! AVAILABLE. ADD CODE HERE TO NOTIFY OPERATOR IF Z8% EXCEEDS USER
! SPECIFIABLE VALUE.

20365    FNM$=CHR$(ERR)
         : RESUME 20390 ! SET VALUE OF FUNCTION TO CHR$ OF ERROR

20370    IF Z1$<>""
         THEN SLEEP 1%
           : GOTO 20340 ! GET REST OF MESSAGE

20380    FNM$=""

20390    ON ERROR GOTO 0 ! MODIFY THIS LINE TO SET ERROR TRAP TO
! USER ERROR ROUTINE

20395    FNEND

32767    END
```

In the next two issue of the RSTS SIG Newsletter, I will publish a proposed standardized message protocal and the results of a bench mark which shows how to optomize inter-task communication to set the most messages per millisecond!

Bob Jones
Interactive Information Systems
P.O. Box 37403
Cincinnati, Ohio 45222

One of the items on the wish list we discussed in Dallas was a function to enable us to derive a printable string from a numeric value greater than 999999. The PRINT USING works fine if you are preparing the value for immediate printout, but not if you simply want to set a string variable to the value of a numeric variable. Here are a couple of function that should do the trick:

```
20000!
!
!       FUNCTION TO RETURN STRING VALUE OF NUMBER
!
!       N IS THE NUMBER, Z IS THE NUMBER OF DECIMAL PLACES
!
20010   DEF FNN1$(N,Z)
 : Z1$="0"
 : Z1=FIX(ABS(N)*10Z+.5) !
20020   IF Z1<>0 THEN Z1$=""
 : Z1%=0%
 : Z1%=0-Z IF Z>0
 : WHILE Z1<>0 OR Z1%<0%
 : Z2%=Z1-FIX(Z1/10)*10
 : Z1=FIX(Z1/10)
 : Z1$=CHR$(48%+Z2%)+Z1$
 : Z1%=Z1%+1%
 : Z1$="."+Z1$ IF Z1%=0%
 : NEXT !
20030   Z1$="-"+Z1$ IF N<0
 : FNN1$=Z1$
 : FNEND !
20040!
!
!       FUNCTION TO RETURN STRING VALUE OF LITERAL
!
20050   DEF FNN7$(N)
 : Z%=LEN(CVT$$(NUM$(ABS(N-FIX(N))),2%))-1%
 : Z1$="0"
 : Z1=FIX(ABS(N)*10Z%+.5) !
20060   IF Z1<>0 THEN Z1$=""
 : Z1%=0%
 : Z1%=0%-Z% IF Z%>0%
 : WHILE Z1<>0 OR Z1%<0%
 : Z2%=Z1-FIX(Z1/10)*10
 : Z1=FIX(Z1/10)
 : Z1$=CHR$(48%+Z2%)+Z1$
 : Z1%=Z1%+1%
 : Z1$="."+Z1$ IF Z1%=0%
 : NEXT !
20070   Z1$="-"+Z1$ IF N<0
 : FNN7$=Z1$
 : FNEND !
```

Bob Jones
Interactive Information Systems
P.O. Box 37403
Cincinnati, Ohio 45222

RSTS Marketplace

DESCRIPTION	CONTACT
1.) A virtual sort subroutine for sorting up to 32,000 records for all files (include virtual) 2.) An easy to use line editor for editing program or ASCII data files. (automatic verification) 3.) A management information system for RSTS 4.) A jobshop scheduler for the allocation of labor and materials in manufacturing or maintenance 5.) A complete general ledger accounting system includes balance sheet, statements, journals and schedules such as accounts payable. 6.) An interactive payroll package with 941 & W-2. 7.) A complete automotive agency accounting system for GM, FORD and other dealers. An on-line system. 8.) A package of real estate programs - includes amortization schedules, data retrieval and others. 9.) Financial package for bidding and printing holdings of municipal and government bonds.	Al Cordes Houston Timesharing 9039 Katy Freeway #404 Houston, Texas 77024 713-461-5437
1.) Subroutines to phonetically code last names for storage and retrieval via standard phonetic techniques. 2.) Screen generator programs for the Tektronix 4023 buffered CRT terminal. 3.) Complete letter writing system using paragraph/letter files to build custom upper/lower case letters from form paragraphs. System allows insertions as well as a user composed paragraph..	Murray Stinson Scientific Placement, Inc. 2700 South Post Oak, #2430 Houston, Texas 77027 713-621-9811
A generalized Data Base and File Management System.	John Wright Florida Computer, Inc. 99 N. W. 183rd St. N. Miami, Florida 33169 305-652-1710
Multiple languages can be used with RSTS/E if you have OMSI's RT-11 emulator. OMSI-RT supports MCRO-11, OMSI APL, and other languages, in addition to DEC FORTRAN IV. 64K words of memory are recommended, but there are several 48K users.	Rusty Whitney OMSI Computing 4015 S.W. Canyon Rd. Portland, Oregon 97221 503-248-5903
1.) A consolidating Accounts Payable, General Ledger, Financial Statement package. 2.) A Payroll and Labor Distribution package. 3.) An Accounts Receivable and Sales Distribution package.	Jack Bouknight Plycom Services P. O. Box 160 Plymouth, Indiana 46563 219-936-2183

RSTS MARKETPLACE (cont)

1) "DATA BOSS" - A completely conversational generalized DBMS with multiple keys; immediate access on massive files; easily customized; contains its own special selection language with logical operators; forces documentation including a Generalized Applications Monitor; contains MULTI-LEVEL PASSWORD protection; allows for network structuring and linked data bases, and unlimited simultaneous users of the same Data Base; requires NO programming to use it; meets the vast majority of the CODASYL Data Base Task group language specifications.

2) COUNTRY CLUB PACKAGE - A complete system including accounts receivable, automatic dues, taxes and gratuities, locker control, cashflow analysis, member spend analysis, optional automatic gratuities calculation by waiter, daily managers report, food and beverage analysis, aging, member progress and activity report, all with complete audit trails.

3) FINSYS - A C.P.A. WRITE UP PACKAGE - Includes Transaction Registers, General Ledger, Multiple Format Profit and Loss Statements, Balance Sheets with comparisons to budget and last year, Employee Earnings Register, 941's, W-2's, Unemployment Reports, Source & Application of Funds. Consolidated and Departmental Statements, Accountants Worksheets.

4) PAYROLL - Includes hourly and automatic salary inputs, input listings & proofs, up to six miscellaneous deductions per employee, Federal, Multiple Local and State Taxes. Complete tips and meals calculations. Report and registers contain Current Pay, Period-To-Date, Quarter-To-Date and Year-To-Date information. Reports include Payroll Register, Check Register, Tax Liability Report, and Labor Distribution Reports, 941's and W-2's.

5) BILL OF MATERIALS EXPLOSION PACKAGE - With Production Planning and Inventory Control for manufacturers.

6) INVENTORY CONTROL SYSTEM - Including Sales Analysis, Reorder points, etc., for wholesalers.

John H. Wright
Florida Computer Inc.
99 N.W. 183 Street
North Miami, Fla 33169
(305) 652-1710

RSTS MARKETPLACE

DESCRIPTION

1.) GAAP - A General Accounting Application Package. A complete, interactive, on-line, multi-user system supporting the areas of general ledger, accounts receivable, accounts payable, and financial reports.
2.) DBM - A general purpose Data Base Management Facility. User defined variable length records, variable length fields, and records with repeating groups. Acceptable characters definable for field values aids in the validation of input data. Direct, sequential, or indexed sequential access are featured.
3.) General Utility Package - Includes programs for file formatting and listing, checking for possible file corruption, reviewing and modifying the contents of disk blocks, restructuring a data base for increased system efficiency, etc.
4.) An on-line, interactive school administration package including advanced scheduling and grade/attendance reporting capabilities.
5.) A commercial timesharing accounting system - Accumulates raw use data (connect hours, CPU time, storage) by system account number. (Such data automatically summarized in month-to-date reports thrice daily.) Applies rates to raw data for monthly billing. Provides for special and flat rates. This package may also be used for non-commercial installation to keep tabs on system utiliazation.

CONTACT

James Degnen
Intelligent Systems Corp.
1 DeAngelo Drive
Bedford, Mass. 01730
617-275-6576

DESCRIPTION

GISMO is an index sequencial access method for users of DEC's RSTS/E (CTS/E) operating system
* Random or sequencial add, delete, and update
* No secearching of overflow chains
* Multiple user support
* Generic keys

CONTACT

Leonard Farano
Gambit Management Strategies
1440 Broadway
New York, NY 10018

LETTERS

Attached is a patch for backup (as if there aren't enough) for use with a TU16. With this patch, a device specification of MTx: causes the drive to be used at the system default density while a specification of MMx: causes it to be used at 1600 bpi.. I offer it only as a quick and dirty solution.

John Ritzenthaler
Coors Porcelain Company
Golden, Colorado 80401
(303) 279-6565 X2159

```
OLD BACKDK

Ready

LIST 710,750,6207
BACKDK  09:42                08-Mar-76
710     IF G1%AND3% THEN 750
        ELSE    OPEN S8$ FOR INPUT AS FILE 1%,RECORDSIZE B1% :
                I%=MAGTAPE(6%,256%,1%)IF LEFT(S8$,2%)="MM" :
                FIELD#1%,B1%ASI$ :
                I%=MAGTAPE(3%,A0%,1%) :
                E%=1% IF S0%
750     IF G3%AND11% THEN 790
        ELSE    GO SUB 6205 IF W%AND4% :
                OPEN S9$ FOR OUTPUT AS FILE 2%,RECORDSIZE B2% :
                I%=MAGTAPE(6%,256%,2%)IF LEFT(S9$,2%)="MM" :
                I%=MAGTAPE(3%,A0%,2%) :
                Y%,Z%=32767% :
                FIELD#2%,B2%AS0$
6207    IF LEFT(S9$,2%)="MM" THEN
                OPEN S9$ AS FILE 1%,MODE 256% :
                I%=MAGTAPE(3%,0%,1%) :
                I%=MAGTAPE(2%,0%,1%) FOR X%=1% TO 3% :
                CLOSE 1% :
                RETURN
```

- CHANGE OF VENUE -

MIKE CROWLEY THE CRUDS RSTS LUG CHAIRMAN HAS HAD CHANGE IN EMPLOYMENT. HIS NEW ADDRESS IS :

MIKE CROWLEY
SENIOR SYSTEMS ANALYST
LUTHERAN GENERAL HOSPITAL
1775 DEMPSTER STREET
PARK RIDGE, ILL. 60068

Here is a suggestion for an improvement in RSTS/E BASIC-PLUS error handling:

ON ERROR GOTO Ø should return error trapping to the last error handling routine as executed in an ON ERROR GATO XXXXXXX

For example:

```
10 ON ERROR GATO 3ØØØØ  :  V$ = SYS(CHR$(6%)+CHR$(-7%)
    ! TO TRAP GLOBAL ERRORS
100 PRINT "AMOUNT";
110 GOSUB 10000  ! TO INPUT AMOUNT
10000  ! SUBROUTINE TO INPUT FINANCIAL TYPE AMOUNTS
10010 ON ERROR GOTO 10100
10020 INPUT AØ$
10080 V = VAL(AØ$)
10090 ON ERROR GOTO Ø  ! SHOULD RETURN ERROR HANLDING TO 30000
10099 RETURN           ! THIS WOULD BE A CHANGE
10100 IF ERR     52 THEN  10090
10110 PRINT "STRANGE AMOUNT.  TRY AGAIN"  : RESUME  10020
```

This change would allow programs that are using the current error handling to operate exactly as they do now. It would also allow us all to write programs with better structure using BASIC-PLUS.

I have in mind that it could be implemented so that

```
ON ERROR GOTO XXX   pushes XXX onto a stack if XXX  Ø
                    pops off the stack if XXX = Ø
```

and when an error occurs control is passed to the line no at the top of the stack.

David S. Cochran
Computing Services
Lake Forest College
Lake Forest, Ill 60045

Problem;:

Variables get clobbered in programs with virtual string arrays and null strings, for example:

NAME: Antonio Didato
FIRM: Michael Reese Medical Center
Clinical Computer Lab.
ADDRESS: Renal Division
2900 S. Ellis
Chicago Illinois ZIP 60616

```
10 E1$=CVT%$(0%) : T8$="ABCDEFGHI"
20 OPEN "AA.A" AS FILE 2%
30 DIM#2%,T$(63%)=64%
35 T$(J%)="" FOR J%=0% TO 63%
40 FOR J%=0% TO 63%
50 D2$=T$(J%)
60 D1$=E1$
70 IF CVT$%(E1$) OR CVT$%(D1$) THEN STOP
80 T1$=LEFT(T8$,1%)
90 PRINT J%, :
        NEXT J% :
        STOP

Ready

run
 0              Stop at line 70

Ready

print ascii(mid(e1$,i%,1%)),for i%=1% to len(e1$)
 65             0
Ready

Print ascii(mid(d1$,i%,1%)),for i%=1% to      len(d1$)
 65             0
Ready
```

NAME: **Harold R. Berenson**

CAMBIT Management Strategies

1440 Broadway
New York, NY ZIP 10018

Problem:

[*,*] PPNs will not work for magtape

Diagnosis:

Direct Chains to PIP to do a magtape directory. Standard PIP will not handle the [*,*] convention properly so Direct does not pass it.

Solution:

Have Direct send the [*,*] PPN since extended PIP will handle it correctly. Users of the older PIP will get garbage but its better then getting nothing at all.

Code:

In Direct change line 4010+1
From: 1$=1$+u$ unless u$= "[*,*]"
To : 1$=1$+u$

Enclosed you will find a copy of a method used by one of our students to break into RSTS/E. We are sending the original copies to Maynard as part of a S.P.R. and giving you a copy to inform you of this "break-in".

In using this method, a non-privileged user can write to tape a BAC file that would need privileged status to run, modifying the tape label to indicate a privileged protection code, and then transfer this file into the public structure with BACKUP.

We are currently running RSTS/E version 6A with standard DEC CUSPS. Although the RSTS/11 user's guide indicates (on page 4-60) that BACKUP will not allow a non-privileged user to transger BAC files, we found out the hard way that this is not the case on our system.

Bob Crawford
Rick Catron
Arizona Western College
P.O. Box 929
(602) 726-1000

Problem: A non-privileged user can gain privileged status on any program in her account.

Diagnosis: The DEC system utility program BACKUP will allow a non-privileged user to transfer a compiled file from magnetic tape to her account and retain whatever protection code was in the tape label. Non-privileged users can write to tape (either with PIP or BACKUP) a .BAC file of their creation that would need privileged status to run, modify the tape label to indicate a privileged protection code, and transfer this file to the public disk structure. This action places the .BAC file in the user's account with the BACKUP retained pribileged protection code. (Attached is an example of this process.) The V6A RSTS-11 System Users Guide states on page 4-60 in its description of BACKUP, "A non-privileged user can not transfer a compiled (run-only) file. If a non-privileged user attempts to transfer a compiled file, BACKUP prints the PROTECTION VIOLATION error message and proceeds to the next file." BACKUP apparently does no checking in this respect...

Cure: Modify BACKUP to run as described in the RSTS-11 System Users Guide. Until the Mod is made BACKUP should be made unavailable to non-privileged users by assigning it a protection code of <124>.

```
HELLO 100,4
Password:
WELCOME TO A.W.I.T.S. RSTS/E TIME SHARING SYSTEM...
TO SEE A SCHEDULE OF HOURS AVAILABLE--JUST TYPE "HOURS"



Ready

DIR/S
 Name .Ext  Size   Prot     Access       Date        Time     Cluster
BUSTIN.BAS     1  < 60>   24-Jan-76   09-Dec-75   12:50 PM   1  SY:[100,4]
MAGMIX.BAS     2  < 60>   24-Jan-76   23-Jan-76   09:25 AM   1

Total of 3 blocks in 2 files in SY:[100,4]


Ready

ASSIGN MT:

Ready

PIP MT:/ZE

REALLY ZERO MT:? Y
MT:  ZEROED

Ready

OLD BUSTIN

Ready

LIST
BUSTIN  03:41 PM         24-Jan-76
10 OPEN "$ACCT.SYS" FOR INPUT AS FILE 1%
20 INPUT LINE #1%, A$
30 & A$;
40 GOTO 20
50 END

Ready

COMPILE

Ready

DIR/S
 Name .Ext  Size   Prot     Access       Date        Time     Cluster
BUSTIN.BAS     1  < 60>   24-Jan-76   09-Dec-75   12:50 PM   1  SY:[100,4]
MAGMIX.BAS     2  < 60>   24-Jan-76   23-Jan-76   09:25 AM   1
TEMP06.TMP     1  < 60>   24-Jan-76   24-Jan-76   03:41 PM   1
BUSTIN.BAC    11  <124>   24-Jan-76   24-Jan-76   03:41 PM   4

Total of 15 blocks in 4 files in SY:[100,4]


Ready
```

```
PIP MT:<BUSTIN.BAC/BL

Ready

CAT MT:
BUSTIN.BAC        11       155     24-Jan-76

Ready

OLD MAGMIX

Ready

LIST
MAGMIX  03:43 PM         24-Jan-76
5 DIM H$(11%)
10 OPEN "MT:" AS FILE 1%, RECORDSIZE 14%
20 FIELD #1%, 14% AS A$
25 I%=MAGTAPE(3%,0%,1%)
30 GET #1%
31 H$=A$+"" \ CLOSE 1% \ OPEN "MT:" AS FILE 2% \ FIELD #2%, 512% AS B$
32 FOR I% = 1% TO 11%
34 GET #2%
36 H$(I%)=B$+""
38 NEXT I%
39 CLOSE 2% \ OPEN "MT:" AS FILE 1% \ FIELD #1%, 14% AS A$
40 LSET A$=LEFT(H$,8%)+CHR$(252%)+RIGHT(H$,10%)
50 I%=MAGTAPE(3%,0%,1%)
100 PUT #1%, COUNT 14%
103 FIELD #1%, 512% AS B$
105 FOR I% = 1% TO 11%
107 LSET B$=H$(I%)
109 PUT #1%
111 NEXT I%
117 I%=MAGTAPE(2%,0%,1%) FOR J% = 1% TO 3%
120 END

Ready

RUN
MAGMIX  03:43 PM         24-Jan-76

Ready

CAT MT:
BUSTIN.BAC        11       252     24-Jan-76

Ready

RUN$BACKUP
FILE BACKUP  V05C-03

$SY:<MT:BUSTIN.BAC/S
TRANSFER COMPLETE 0 BLOCKS TRANSFERED IN 1 FILES

$^Z

Ready
```

```
DIR/S
 Name .Ext  Size    Prot      Access        Date        Time     Cluster
BUSTIN.BAS     1   < 60>   24-Jan-76   09-Dec-75   12:50 PM    1   SY:[100,4]
MAGNIX.BAS     2   < 60>   24-Jan-76   26-Jan-76   09:25 AM    1
TEMP06.TMP     2   < 60>   24-Jan-76   24-Jan-76   03:41 PM    1
BUSTIN.BAC    11   <252>   24-Jan-76   24-Jan-76   03:44 PM    1

Total of 16 blocks in 4 files in SY:[100,4]


Ready

RUN BUSTIN
1,0,PACMFD,0,0,MASTER FILE DIRECTORY
1,2,AWCLIB,0,0,AWCTEM LIBRARY
1,3,AUXLIB,0,4,AWC UTILITIES NOT DEC SUPPORTED
1,4,LOCK,0,4,FUN AND GAMES
1,5,CAI,0,4,APPLICATION PROGRAMS
1,6,MANAGR,0,4,MGR ACCT PRI BAK FILES
1,7,LOVEIT,0,0,RICK CATRON
1,8,AWCBOB,0,0,BOB C PRIV
1,9,AWCTOM,0,0,TOM SCHELLHASE
1,10,AWCRON,0,0,RON COSTIN
1,11,ZIPO,0,0,RUSTY'S MESS...
1,12,WAYIN,0,0,RICHARD DRINKARD
1,13,MANR,0,4,MANR STUFF
1,14,MANR2,0,4,MANR STUFF CONT
1,15,MANR3,0,4,MANR STUFF CONT MORE
1,16,AWCNOR,0,0,NORM RIEBE
1,121,MAINT,0,0,DEC MAINT A^C

Ready

BYEF
```

CHANNEL 13

April, 1976

Well, we got a gratifying return from last time's questionnaire, not only in ratings, but in comments, observations, and new suggestions as well. Thanks to everyone who took time to respond, and of course, special thanks to each of you who offered your thoughts for the next go-around.

I'd like to reflect a little on how things seem to be developing. It's becoming clear that our suggestions for RSTS' future are not all of one type: elements of design philosophy which need "steering" (in our estimation, of course); new areas of functionality; there are problems needing correction; and finally, logical extensions and improvements. Examples in each area might be Basic-Plus compatibility, batch processing, a "reconstructor" program, and double-precision integers, respectively (see the last Ch. 13).

It is important that we try to do justice to the first two of these areas, where a dialogue such as this may have the maximum effect in persuading DEC of the needs in the user community and the maximum utility in promoting the development of a yet more saleable product. However, the best contributions to design philosophy and new functionality are likely to result either from suggestions of general direction or from someone's offering a unified sketch of a possible implementation. An example of the latter are the thoughts on BATCH offered by Messrs Weyrick, Metcalfe, and Jones in this issue. This kind of thing probably <u>is</u> best handled by actual letters and reports from the field, as opposed to a turnaround questionnaire.

There are many members of our community who have found it necessary to "roll their own" to get the job done. That the need exists hardly surprises the rest of us, and DEC, too, is undoubtedly aware of the activity. What we need to do, though, is endeavor to make as much of our experience as we can available in useful form to nurture the hope of eventual support by standard DEC offerings. So if you've had to roll <u>your</u> own, or if you're contemplating the need, send in the ideas.

The last newsletter (and the meeting in Dallas) generated a considerable body of material, both of this type (BATCH, accounting and backup, particularly) and of the rank-by-questionnaire type. Of course, DEC may not respond to every suggestion, nor (obviously) even be able to. However, it would be a shame for necessities to be invented repeatedly in the field just because Maynard hadn't heard. So send in ideas; those that fit will go in the questionnaire to get whatever weight the membership can command; the broad ones will be collected and sent on to the developers at DEC. We will report on them all in the next Ch. 13.

Now: results from last time: We seem by far to be most concerned about compatibility between Basic-Plus'es (see the ranking sheet). Some responses even indicated "extra-vital", and included scribbled emphasis. As we heard in Dallas, there are already people within DEC who themselves feel that compatibility is high-priority, but it's far from clear what disposition of the issue will finally be made by corporate policy.

One-way (upward) compatibility <u>is</u> planned, and B+ II will come with a translator program. This should pretty much take care of those installations planning to simply switch over; the most likely source of trouble will be like

```
PRINT USING A$, B%
```

since the translator won't be able to determine the object-time value of the edit string to reformulate it.

(Parenthetically, in 6B: "&" will no longer be legal for PRINT; ":" will no longer serve as a statement separator - use backslash instead. This applies to current B+ as well as B+ II!)

The real issue of compatibility may turn out to be more than one-way, however, if users' decisions about buying B+ II are strongly influenced by debugging facilities. B+ II is planned to have things such as breakpoints, and the compile time is supposed to be approximately like B+ I "OLD" (interpreter) time, but specs haven't been finalized on how dynamic the debug facilities will be, or indeed, exactly what will be included. There are some things that can't be: incremental compiling, for instance (modifying a program during a STOP).

There would be less potential problem if B+ I and B+ II evolved together and had the same processing capabilities. Then one could always drop back to "old" B+, slower though it be, to isolate some nasty bug. If, however, users buying B+ II to get Record Management support, for instance, can't use their new capabilities under "old" B+, they will be totally dependent on the forthcoming (and somewhat different) B+ II debugging aids.

At this writing, DEC's plans seem to suggest that B+ II won't be downward compatible, and that the switch will be one-way for users needing the new goodies. What we should now try to supply to DEC planners is some idea about the extent to which we see the marketability of the "new" Basic-Plus being tied to its debugging facilities, however they are obtained. Clearly, the questionnaire response shows that users feel it's a paramount issue, and we can only wish there were some way to establish its relationship to dollars and cents for DEC.

- - - - - - - - - -

Item ¿8? on the questionnaire (Multiple Buffer Assignment) has moved over from Questions to Answers: the word at Dallas was that this will be a feature in 6B.

- - - - - - - - - -

Item ¿1? ("Date Last Modified" file statistics) got a strong vote: 63.2% of those responding would be willing to replace "Creation Time"; another 12.3% would replace "Creation Date"; 19.3% said "forget it".

- - - - - - - - - -

We've divided the other items into "RSTS-related" and "B+-related" categories (see the ranking sheet). On the whole, extensions to RSTS seemed to rate higher than extensions to B+, and to "clump" more. Differences within clumps are likely to mean little.

Statistics: 57 questionnaires were returned. Weighting: Indifferent - 0; Merely Nice - 1; Important - 2; Vital - 4; Extra Vital (marginal notes) - 5.

Finally: these results, of course, reflect only a portion of the RSTS user community, but we might hope that it's a concerned and reflective group. And relative

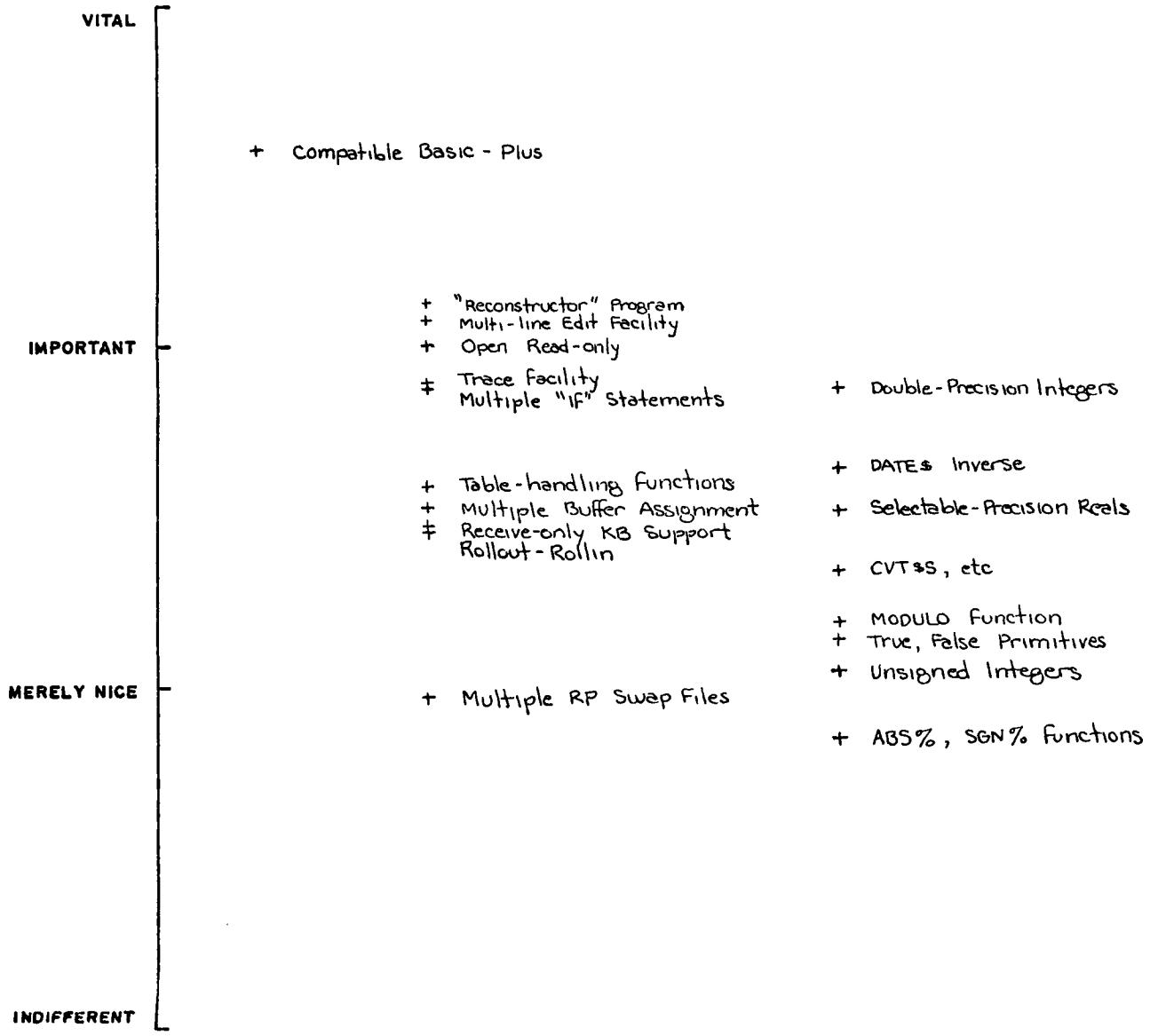
rankings shouldn't suggest ignoring ideas which placed low: nothing got all zeros, and some things might even be easy for DEC to implement.

- - - - - - - - - - -

See you next time, and in Atlanta

Bill Sconce, University of Louisville HSCC
P. O. Box 1055, Louisville, KY 40201
(502) 588-5408

RANKING - JANUARY QUESTIONNAIRE



Thanks to COSAP for making statistics almost fun.

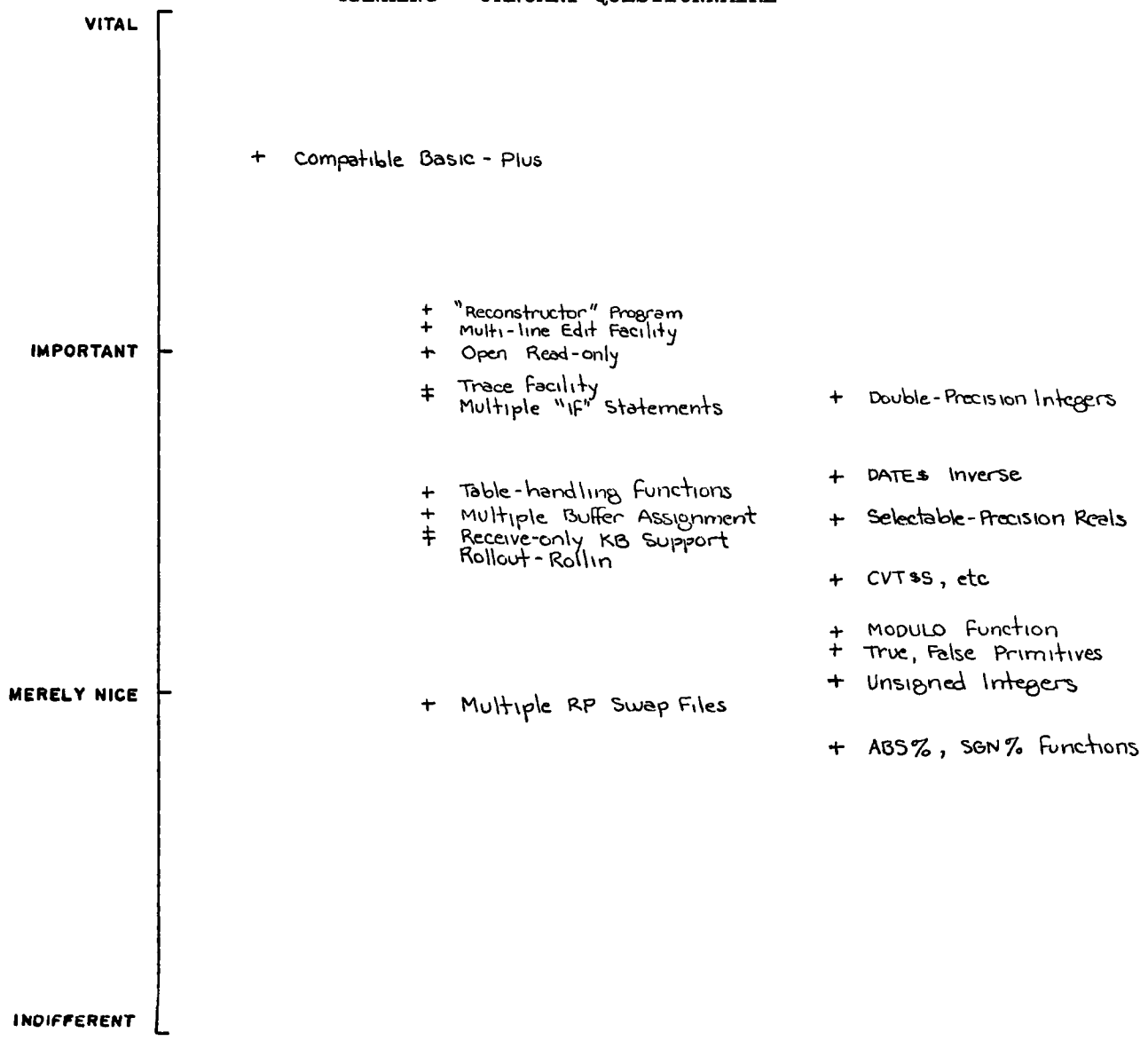
rankings shouldn't suggest ignoring ideas which placed low: nothing got all zeros, and some things might even be easy for DEC to implement.

- - - - - - - - - - -

See you next time, and in Atlanta

Bill Sconce, University of Louisville HSCC
P. O. Box 1055, Louisville, KY 40201
(502) 588-5408

RANKING - JANUARY QUESTIONNAIRE



Thanks to COSAP for making statistics almost fun.

Accounting data read/read-reset	7-90	14
Accounting dump	7-93	-15
Assign/reassign device	7-30	10
Attach	7-77	6
Broadcast to a terminal	7-41	-5
CTRL/C trap enable	7-36	-7
Clean up a disk pack	7-48	2
Date/time changer	7-39	-14
Deassign all devices	7-33	12
Deassign device	7-32	11
Detach	7-75	7
Directory lookup by filename: disk	7-100	17
Directory lookup magtape	7-97	15
Directory lookup on index	7-95	15
Directory lookup wildcard: disk	7-101	17
Disk cache enable/disable	7-109	19
Disk pack mount, etc /TTYSET	7-45	3
Drop temporary privileges	7-59	-21
Error message decode	7-29	9
FCB/DDB information	7-108	-8
File backup statistics change	7-70	-11
Filename scan	7-20	-10
Filename scan, terminating	7-20	-23
Force input to a terminal	7-42	-4
Hangup a dataset	7-40	-9
Kill job	7-51	8
Lock/unlock job in core	7-58	-20
Login	7-72	4
Logins disable	7-43	-2
Logins enable	7-44	-1
Logins set	7-89	-19
Logout	7-74	5
Message receive	7-82	18
Message send	7-85	18
Monitor tables I	7-105	-3
Monitor tables II	7-107	-12
PPN create	7-60	0
PPN delete	7-62	1
Password/quota change	7-49	8
Poke core	7-88	-6
Priority/runburst/jobsize	7-54	-13
Quota/password change	7-49	8
Reassign/assign device	7-30	10
Reattach	7-79	6
Receive message	7-82	18
Receiver declare	7-82	18
Receiver remove	7-87	18
Run-time system functions	7-111	-18
Run-time system name	7-110	-17
Send message	7-85	18
Shutup	7-38	-16
Special run priority	7-56	-22
TTY disable	7-52	8
TTYSET	7-64	16
TTYSET/disk pack stuff	7-64	3
Zero a device	7-34	13

Accounting data read/read-reset	7-90	14
Accounting dump	7-93	-15
Assign/reassign device	7-30	10
Attach	7-77	6
Broadcast to a terminal	7-41	-5
CTRL/C trap enable	7-36	-7
Clean up a disk pack	7-48	2
Date/time changer	7-39	-14
Deassign all devices	7-33	12
Deassign device	7-32	11
Detach	7-75	7
Directory lookup by filename: disk	7-100	17
Directory lookup magtape	7-97	15
Directory lookup on index	7-95	15
Directory lookup wildcard: disk	7-101	17
Disk cache enable/disable	7-109	19
Disk pack mount, etc /TTYSET	7-45	3
Drop temporary privileges	7-59	-21
Error message decode	7-29	9
FCB/DDB information	7-108	-8
File backup statistics change	7-70	-11
Filename scan	7-20	-10
Filename scan, terminating	7-20	-23
Force input to a terminal	7-42	-4
Hangup a dataset	7-40	-9
Kill job	7-51	8
Lock/unlock job in core	7-58	-20
Login	7-72	4
Logins disable	7-43	-2
Logins enable	7-44	-1
Logins set	7-89	-19
Logout	7-74	5
Message receive	7-82	18
Message send	7-85	18
Monitor tables I	7-105	-3
Monitor tables II	7-107	-12
PPN create	7-60	0
PPN delete	7-62	1
Password/quota change	7-49	8
Poke core	7-88	-6
Priority/runburst/jobsize	7-54	-13
Quota/password change	7-49	8
Reassign/assign device	7-30	10
Reattach	7-79	6
Receive message	7-82	18
Receiver declare	7-82	18
Receiver remove	7-87	18
Run-time system functions	7-111	-18
Run-time system name	7-110	-17
Send message	7-85	18
Shutup	7-38	-16
Special run priority	7-56	-22
TTY disable	7-52	8
TTYSET	7-64	16
TTYSET/disk pack stuff	7-64	3
Zero a device	7-34	13

<1>	[99,99]	no read
<2>	[99,99]	no write
<4>	[99,x]	no read
<8>	[99,x]	no write
<16>	others	no read
<32>	others	no write
<64>	-	compiled
<128>	-	privileged

Libraries:

$ = [1,2]
! = [4,3]
% = [4,4]
& = [4,5]
# = [99,0]
@ = assignable

- cvt$$(S$,M%) -

M% = 1% trim parity bit
M% = 2% discard all spaces and tabs
M% = 4% discard CR, LF, FF, ESC, RUB, NUL
M% = 8% discard leading spaces and tabs
M% = 16% reduce spaces & tabs to one space
M% = 32% lower case to upper case
M% = 64% [] to ()
M% = 128% discard trailing spaces and tabs

- Functions -

+	string concatenation	B+ 5-13
ascii(X$)	ascii value of first character in X$	B+ 5-13
chr$(N%)	one-char string having ascii value of N%	B+ 5-13
date$(N%)	99-Mmm-99 where N% = julian day + 1000(years since 1970)	B+ 8-24
date$(0%)	99-Mmm-99 current date	B+ 8-24
instr(N1%,A$,B$)	seek B$ within A$ beginning @ position N1%	B+ 5-14
left(A$,N%)	leftmost N% characters of A$	B+ 5-13
len(A$)	number of characters in A$, incl trailing spaces	B+ 5-13
mastape(1%,0%,n%)	rewind & off-line	
mastape(2%,0%,n%)	write tapemark	
mastape(3%,0%,n%)	rewind	All: n% = channel
mastape(4%,m%,n%)	skip m% records	RSTS/E Programming, section 2.8
mastape(5%,m%,n%)	backspace m% records	
mastape(6%,x%,n%)	set density/parity	
mastape(7%,0%,n%)	get mastape status 2-17	
mid(A$,N1%,N2%)	substring of A$, beginning at N1%, N2% characters long	B+ 5-13
num$(N)	<sp> <value of N> <sp> (as would be output by PRINT stmt)	B+ 5-14
rad$(I%)	three-char string converted from radix-50	B+ 8-25
right(A$,N%)	rightmost part of A$, beginning at position N%	B+ 5-13
space$(N%)	string of N% spaces	B+ 5-14
status	see B+ 12-2	
string$(N1%,N2%)	string of length N1%, characters of ascii decimal value N2%	B+ 5-15
swap%(I%)	word I% returned with its bytes swapped	B+ 8-25
time(0)	clock time in seconds since midnight	
time(1)	CPU time for job in 0.1 second quanta	
time(2)	connect time in minutes	All: B+ 8-24
time(3)	KCT's used	
time(4)	device time in minutes	
time$(N%)	string corresponding to time at N minutes before midnight	
time$(0%)	string corresponding to current time	
val(A$)	value of string as would have been returned if A$ had been typed in response to INPUT	B+ 5-14

<1>	[99,99]	no read
<2>	[99,99]	no write
<4>	[99,x]	no read
<8>	[99,x]	no write
<16>	others	no read
<32>	others	no write
<64>	-	compiled
<128>	-	privileged

Libraries:

$ = [1,2]
! = [4,3]
% = [4,4]
& = [4,5]
# = [99,0]
@ = assignable

- cvt$$(S$,M%) -

M% = 1% trim parity bit
M% = 2% discard all spaces and tabs
M% = 4% discard CR, LF, FF, ESC, RUB, NUL
M% = 8% discard leading spaces and tabs
M% = 16% reduce spaces & tabs to one space
M% = 32% lower case to upper case
M% = 64% [] to ()
M% = 128% discard trailing spaces and tabs

- Functions -

+	string concatenation	B+ 5-13
ascii(X$)	ascii value of first character in X$	B+ 5-13
chr$(N%)	one-char string having ascii value of N%	B+ 5-13
date$(N%)	99-Mmm-99 where N% = julian day + 1000(years since 1970)	B+ 8-24
date$(0%)	99-Mmm-99 current date	B+ 8-24
instr(N1%,A$,B$)	seek B$ within A$ beginning @ position N1%	B+ 5-14
left(A$,N%)	leftmost N% characters of A$	B+ 5-13
len(A$)	number of characters in A$, incl trailing spaces	B+ 5-13
mastape(1%,0%,n%)	rewind & off-line	
mastape(2%,0%,n%)	write tapemark	
mastape(3%,0%,n%)	rewind	All: n% = channel
mastape(4%,m%,n%)	skip m% records	RSTS/E Programming, section 2.8
mastape(5%,m%,n%)	backspace m% records	
mastape(6%,x%,n%)	set density/parity	
mastape(7%,0%,n%)	get mastape status 2-17	
mid(A$,N1%,N2%)	substring of A$, beginning at N1%, N2% characters long	B+ 5-13
num$(N)	<sp> <value of N> <sp> (as would be output by PRINT stmt)	B+ 5-14
rad$(I%)	three-char string converted from radix-50	B+ 8-25
right(A$,N%)	rightmost part of A$, beginning at position N%	B+ 5-13
space$(N%)	string of N% spaces	B+ 5-14
status	see	B+ 12-2
string$(N1%,N2%)	string of length N1% characters of ascii decimal value N2%	B+ 5-15
swap%(I%)	word I% returned with its bytes swapped	B+ 8-25
time(0)	clock time in seconds since midnight	
time(1)	CPU time for job in 0.1 second quanta	
time(2)	connect time in minutes	All: B+ 8-24
time(3)	KCT's used	
time(4)	device time in minutes	
time$(N%)	string corresponding to time at N minutes before midnight	
time$(0%)	string corresponding to current time	
val(A$)	value of string as would have been returned if A$ had been typed in response to INPUT	B+ 5-14

<1>	[99,99]	no read
<2>	[99,99]	no write
<4>	[99,x]	no read
<8>	[99,x]	no write
<16>	others	no read
<32>	others	no write
<64>	-	compiled
<128>	-	privileged

Libraries:

$ = [1,2]
! = [4,3]
% = [4,4]
& = [4,5]
# = [99,0]
@ = assignable

- cvt$$(S$,M%) -

M% = 1% trim parity bit
M% = 2% discard all spaces and tabs
M% = 4% discard CR, LF, FF, ESC, RUB, NUL
M% = 8% discard leading spaces and tabs
M% = 16% reduce spaces & tabs to one space
M% = 32% lower case to upper case
M% = 64% [] to ()
M% = 128% discard trailing spaces and tabs

- Functions -

+	string concatenation	B+ 5-13
ascii(X$)	ascii value of first character in X$	B+ 5-13
chr$(N%)	one-char string having ascii value of N%	B+ 5-13
date$(N%)	99-Mmm-99 where N% = julian day + 1000(years since 1970)	B+ 8-24
date$(0%)	99-Mmm-99 current date	B+ 8-24
instr(N1%,A$,B$)	seek B$ within A$ beginning @ position N1%	B+ 5-14
left(A$,N%)	leftmost N% characters of A$	B+ 5-13
len(A$)	number of characters in A$, incl trailing spaces	B+ 5-13
mastape(1%,0%,n%)	rewind & off-line	
mastape(2%,0%,n%)	write tapemark	
mastape(3%,0%,n%)	rewind	All: n% = channel
mastape(4%,m%,n%)	skip m% records	RSTS/E Programming, section 2.8
mastape(5%,m%,n%)	backspace m% records	
mastape(6%,x%,n%)	set density/parity	
mastape(7%,0%,n%)	get mastape status 2-17	
mid(A$,N1%,N2%)	substring of A$, beginning at N1%, N2% characters long	B+ 5-13
num$(N)	<sp> <value of N> <sp> (as would be output by PRINT stmt)	B+ 5-14
rad$(I%)	three-char string converted from radix-50	B+ 8-25
right(A$,N%)	rightmost part of A$, beginning at position N%	B+ 5-13
space$(N%)	string of N% spaces	B+ 5-14
status	see B+ 12-2	
string$(N1%,N2%)	string of length N1%, characters of ascii decimal value N2%	B+ 5-15
swap%(I%)	word I% returned with its bytes swapped	B+ 8-25
time(0)	clock time in seconds since midnight	
time(1)	CPU time for job in 0.1 second quanta	
time(2)	connect time in minutes	All: B+ 8-24
time(3)	KCT's used	
time(4)	device time in minutes	
time$(N%)	string corresponding to time at N minutes before midnight	
time$(0%)	string corresponding to current time	
val(A$)	value of string as would have been returned if A$ had been typed in response to INPUT	B+ 5-14

This new RSTS user profile was developed by Elizabeth Matlack of Dow Chenical and Howard Kipple of New Mexico Junior College. Please send your completed profile to:

Linda Cohen
First National City Bank
485 Lexington Ave.
New York, NY 10017

RSTS USER PROFILE

Date of Installation: _____

Contact Person _____

Address: _____

City, State, Zip _____

Company _____ Phone _____

For multiple installations please reporduce this form and respond for each installation.

Respond with quantity of each item.

Processor: 11/20 _____ 11/40 _____ 11/45 _____ 11/50 _____ 11/70 _____ Other _____

Operating System: V4A ____ V5A _____ V5B _____ V5C _____ V6a _____ V6B ____ Other _____

Languages: COBOL _____ Basic + _____ Fortran _____ Dibol _____ Other _____

Memory Size: MOS _____ Cache _____ Core _____ Bipole _____ Other _____

Vendor of other _____

RJE 2780 Corrected to: 360 ____ 370 _____ Other _____

Card Reader: CR11 _____ CM _____ Other _____

Card Punch: _____

Line Printer: LS 11 _____ LP 11 _____ Other _____

Terminals: TTY _____ LA 30 _____ LA 36 _____ Other _____

Vendor _____.

CRTS: VT 05 _____ VT 50 _____ VT 52 _____ Other _____ Vendor _____

Lines: DH 11 _____ DN 11 _____ DL 11/A _____ DL 11/E _____ Other _____

Disk Drives: RK 02 _____ RK 05 _____ RP 02 _____ RP 03 _____ RP 04 _____

RS _____ RF _____ Other _____ Vendor _____

Mag Tape: 7 Track _____ TU 10 _____ TU 16 _____ 9 Track _____

Other _____ Vendor _____

DECTAPE: TU 56 _____

Floppy Disk _____

Current Applications: _____

_____

Need Software For: _____

_____

_____